Class Activation Map Analysis of Deep Networks for Pokémon Classification

Shivani Naik
College of Engineering
Northeastern University
Seattle, WA, United States
naik.shiv@northeastern.edu

Pulkit Saharan
College of Engineering
Northeastern University
Seattle, WA, United States
saharan.p@northeastern.edu

***Abstract*—With time, one particular algorithm— Convolutional Neural Network—has been developed and optimized, primarily leading to breakthroughs in computer vision with deep learning. A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning system that can take in an input image, assign importance (learnable weights and biases) to various elements and objects in the image, and be able to distinguish between them. Class Activation Mapping is a technique that can be used to get the visual explanation of the prediction of CNN. In this study, we analyze the Class Activation Maps of pre-trained deep convolutional neural networks to classify Pokémon. We used the VGG16 and AlexNet networks to assess the class activation map as well as VGG19 network for Grad-CAM. We successfully used class activation maps to examine the discriminative region of an image using the seven-category Pokémon dataset from Kaggle.**

***Index Terms*—class activation maps, computer vision, CNN, image, CAM, GRAD-CAM**

I. Introduction

Image classification is crucial to computer vision and is significant for our studies, work, and everyday lives. The method of classifying an image involves picture preprocessing, image segmentation, key feature extraction, and matching identification. Using Convolutional architectures in image classification is a crucial component of deep learning. In the last decade, Convolutional Neural Networks have given ground breaking results in the field of computer vision, pattern recognition, image processing and many more. Convolutional Neural Networks use convolutional architecture to extract the features of images and use fully connected layers for classification [7]. Along with classification of images, we can also see the important region of an image which contributes highly to the prediction by simply tweaking some of the layers of the network. Class Activation Mapping is a technique which provides the visual reason behind the prediction of a particular category by CNN. As shown in Fig. 1, tweaked CNN trained on the Pokemon dataset able to highlight the discriminative region of image for classification. We lose the ability of CNN to show the localized object in the convolutional layer by feeding the feature maps into fully connected layer to do classification. In order to visualize the local discrimative region of an image, global average pooling layer is used in place of fully-connected layer. This is done in order to maintain the spatial information contained in the output of the last convolution layer [1]. In addition to aiding in the visualization of class activation maps, the global average pooling layer also aids in preventing overfitting because it lacks any parameters that can be optimized.

Fig. 1. Class Activation Map showing important regions of an image which allows the CNN to classify the images

II. Related Work

A. Class Activation Maps

Class Activation Mapping [1] represents the discriminative image region of a particular category, utilized by Convolutional Neural Network to identify that category and produces visual explanation maps. Class Activation Maps provide a visual explanation of the neural network prediction. Class activation mapping allows you to determine whether a particular area of an input image "confused" the network and caused it to predict incorrectly.

B. Learning Deep Features for Localization

The most similar approach to ours is the work based on global average pooling [1], in each of the networks they remove the fully-connected layers before the final output and replace them with global average followed by a fully-connected softmax layer. They noted that although the method they present here, global average pooling, is not new, to their knowledge, it is the only one that can be used for precise discriminative localization.

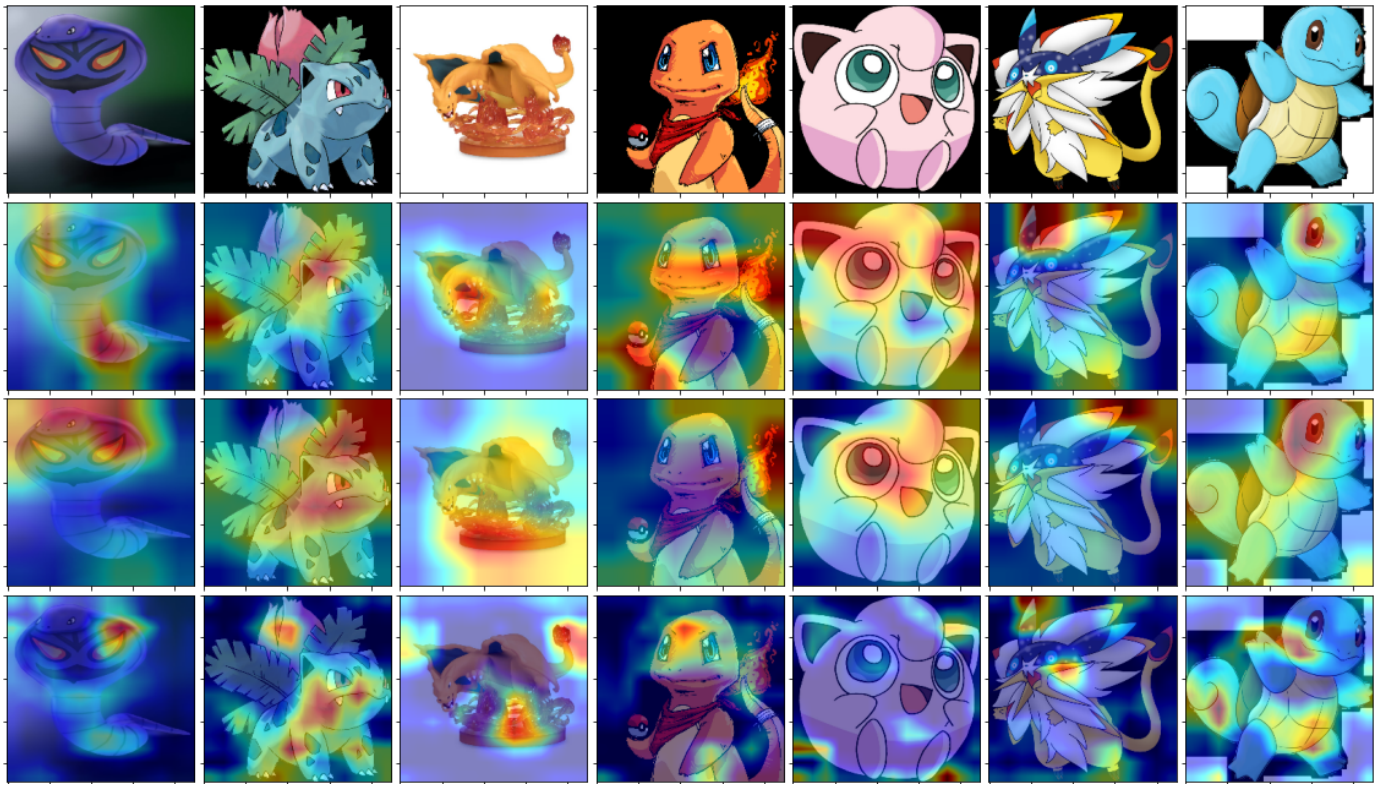

Fig. 2. Discrimative regions for seven categories of Pokemon are shown in this figure. First row shows the original image, second row shows the informative regions from VGG16 CAM network, third row shows the informative regions from AlexNet CAM network and forth row shows the informative regions from VGG19 GRAD-CAM network. Pokemon categories from left to right - Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle.

The global max pooling achieves similar classification performance as global average pooling, global average pooling outperforms global max pooling for localization. They discovered that networks' capacity for localization increased when the final convolutional layer prior to global average pooling contained a better mapping resolution, which is a measure of spatial resolution. To accomplish this, they eliminated many convolutional layers from some of the networks. Both GoogLeNet-GAP and GoogLeNet perform much better than AlexNet. Additionally, despite the latter having less convolutional layers, it was seen that GoogLeNet-GAP and GoogLeNet perform similarly. Overall, it was discovered that GoogLeNet-GAP characteristics were on par with cutting-edge generic visual features.

C. Relevance weighted CAM

As we know majority of the approaches to get the class activation map focused on examining the activation maps of the final convolutional layer because it is well known that this layer possesses high-level semantics. Additionally, the noisy and non-class specific nature of the heatmaps of intermediate layers produced by the earlier techniques further supported this approach. In a paper by Lee, Kim, Park, Eo and Hwang [2], they focused on the layers of shallower depth and able to obtain meaningful information. They came to the conclusion that even in shallow layers with limited receptive fields, class-specific features may be retrieved. Their Relevance- CAM approach uses Layer-wise Relevance Propagation which, using a particular relevance propagation rule, distributes the model class output scores into the input image. Their result shows that the Relevance-CAM's ability to extract class specific features from intermediate and shallow layers is insufficient and ResNet50 outperforms VGG16 with Relevance-CAM.

D. Grad-CAM

Grad-CAM [3] improves the generalizability of CAM, making it possible to produce class activation maps for any pre-built CNN-based image classifier. Grad-CAM makes use of a feature map's average gradients to demonstrate its significance to the intended audience. These techniques can all successfully locate the target objects, but they all share the drawback of relying on the final convolutional layer of CNN to provide class activation maps. Due to the final output's lackluster spatial resolution, the resulting class activation maps can only find broad object regions because to the convolutional layer.

III. Methodology

We took a guided and efficient approach when implementing pre-trained networks to visualise the class activation maps for different categories. In the Torchvision package, a number of pre-trained networks are available. We looked at the pre-trained networks and picked four that were relevant to our project. In our work, we chose to examine the VGG16, Alexnet

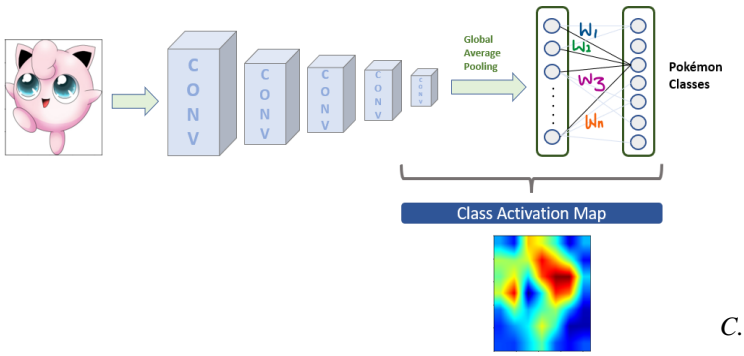

Fig. 3. Class Architecture Map Architecture

and VGG19 networks' differing approaches to feature recognition and discriminative localization and dropped ResNet because of bad performance. We began by importing the pre-trained models, analyzing their architecture, and then fine-tuning them to construct the Class Activation Map Model. Retraining the models on our Pokemon dataset allowed us to proceed and obtain the new weights.

A. Dataset

The dataset utilized in this experiment is publicly accessible on Kaggle.com. The collection includes 151 categories of Pokémon, with anything between 60 and 300 photos in each category. The seven Pokémon categories included in the analysis are Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle. Each category's dataset was divided into train and test portions at random, yielding 1201 training images and 269 validation images. Every image in the collection had a distinct original size. The training and testing photos were scaled to 224x224 pixels and normalized to the RGB mean.

B. Network Architecture

- We began by creating the Class Activation Map network using the pre-trained architecture of VGG16 and Alexnet. The last convolutional layer of VGG 16 gives an output shape of [512,7,7] where 512 is the dimension and 7x7 is the height and width. We replaced the fully connected layer with global average pooling layer. Therefore we have to reshape the last convolutional layer to get our output class. There we need to convert 512,7,7 to 512,49 from there we converted it to 512x1 matrix where 512 is the number of rows and 1 is the number of columns, we can achieve this by doing mean(1).
- The Class Activation Map network architecture for Alexnet was built using the same approach. The final convolutional layer was scaled down from [256,6,6] to 256x1. We can see in Fig. 3, the architecture by which an input image is passed through several convolutional layers and then connected to the final classification layer through a global average pooling layer to produce a class activation map. The global average pooling layer feeds the classification layer directly with the feature vector it receives by averaging each feature map at the final convolutional layer.
- We need to transform 512x1(256x1) to 1x512(1x256) for VGG16(Alexnet) network and apply a linear layer, which will accept 512(256) input features and output 7 number of features, in order to obtain our 7 classes. The output features should be equal to the number of classes we have in our dataset.

C. Model Training

The VGG16 and Alexnet networks were modified to form the CAM architecture. To obtain the new weights for the VGG16 and Alexnet CAM models, we continued and trained both models using our training data. On training data collected over 10 epochs with batch size equal to 1, we solely trained the weights for the new fully connected layer. We used stochastic gradient descent optimizer with learning rate of 0.001 and momentum of 0.9. In the training process, we also used learning rate scheduler to adjusts the learning rate between the epochs. Cross Entropy Loss function has been used in all the network training. **For Grad-CAM,** we changed the number of output class in pre-trained VGG19 network and trained it on our training set. To implement the Grad-CAM, we dissected the network by finding it's last convolutional layer (layer number 36) and rescontructed the VGG19 network. By reconstructing the network, we registered the hook at the last layer of the feature block of our network to pull the gradients out of the model before they are discarded.

D. Prediction and Visualization

Following model training, we used the softmax method to predict the class of a Pokemon and extracted the weights for fully connected layer with shapes of (7,512) for VGG16 and (7,256) for AlexNet. In order to create the color-coded heatmap known as the class activation map, we pulled the feature maps from the network's final convolution layer together with the predicted class score and weights. In order to identify the region in the image, we blend the heatmap with the original image after creating the CAM, which shows the discriminative zone in red. As seen in Fig. 5, the first row displays the original image for each of the seven types of Pokémon, the second row displays the heatmap created using CAM, and the third row displays the blended image where we can finally identify the significant area marked in red.

IV. Experiments and Results

The experiments we conducted for this project and their outcomes are detailed in this section. As stated in the approach, we generated CAM and GRAD-CAM for this project using three different CNN networks. The next subsections of each network provides a detailed explanation of the classification and class activation map results.

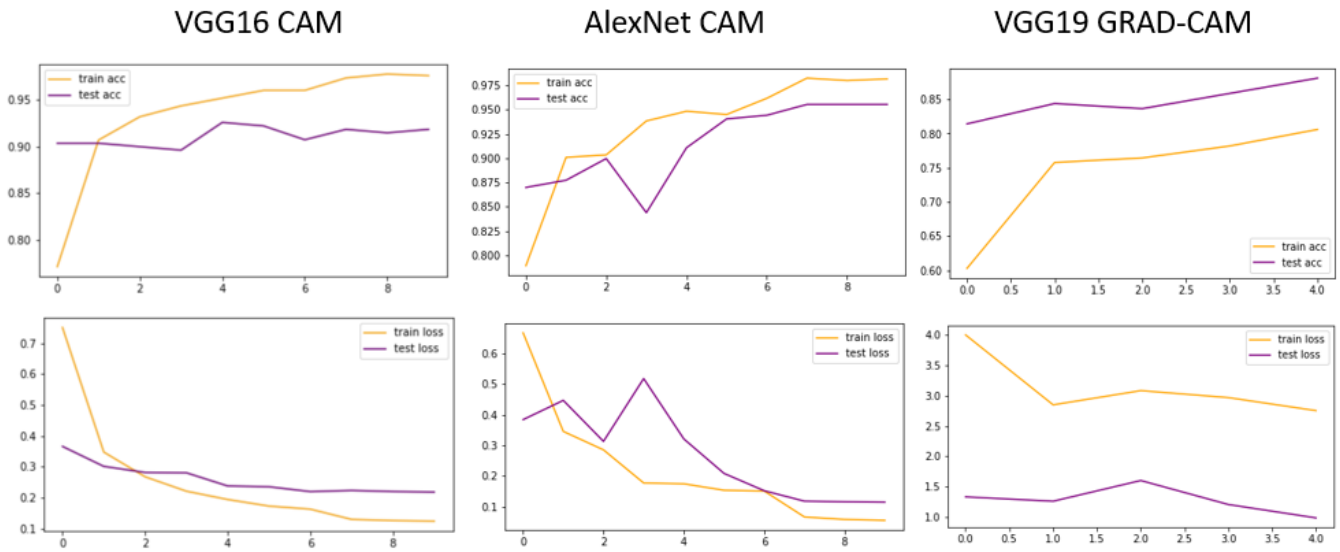

Fig. 4. This figure shows the loss and accuracy plots of all the three networks. Top row from left to right - Train and Test Accuracy plot for VGG16 CAM Network, AlexNext CAM Network and VGG19 Grad-CAM Network. Bottom row from left to right - Train and Test Loss plot for VGG16 CAM Network, AlexNext CAM Network and VGG19 Grad-CAM Network.

Deep CNN Networks			**Loss**		**Accuracy**	
	Batch Size	***Epochs***	***Train***	***Test***	***Train***	***Test***
VGG16 (CAM)	1	10	0.1239	0.2181	0.9759	0.9182
Alexnet (CAM)	1	10	0.0553	0.1146	0.9817	0.9554
VGG19 (GRAD-CAM)	1	10	2.7536	0.9843	0.8060	0.8810

TABLE I
COMPARITIVE EVALUATION OF VGG16 CAM, ALEXNET CAM AND VGG19 GRAD-CAM NETWORK

A. VGG16 CAM

We trained the modified network on our training dataset after modifying the final convolutional layer by including a global average pooling layer after the final convolutional layer. The train and test losses from the VGG16 CAM network are displayed in the bottom left graph of Fig. 4. After 10 epochs, the model's train loss drops from 0.7 to 0.12, and the test loss reaches a value of 0.2181. The same figure shows that test accuracy reaches a value of 0.92 and train accuracy reaches a maximum of 0.97 after 10 epochs. CAM generated from VGG16 shows satisfying result as seen in Fig. 5, the image grid shows the original image Pokemon from each of the seven categories with their heat maps and blended image showing the informative region in the image. It is visible from the image how CAM is showing the wings of Charizard as important region and similarly face of Bulbasuar is identified as an important region for prediction. In three separate images from each category of Pokémon, the instructive sections are depicted in Fig. 6. Here, too, we can see that Pikachu's ears—the most significant area in each image—are highlighted in red, just like the Squirtle's shell is—as well as other Pokémon.

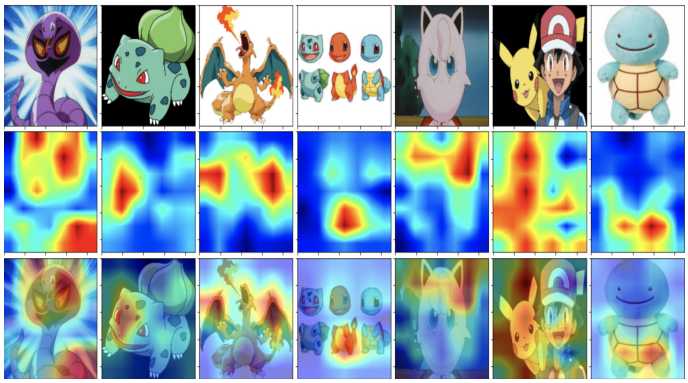

Fig. 5. Class Activation Map of seven categories of Pokemons using VGG16 CAM network. First row- Original Image, second row- Heatmap, third row- Blended image. Pokemon categories from left to right - Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle.

B. AlexNet CAM

After changing the final convolutional layer by adding a global average pooling layer following the final convolutional layer, we trained the redesigned network using our training dataset. The mid bottom graph of Fig. 4 shows the train and test losses for the AlexNet CAM network. The model's train loss falls from 0.6 to 0.05 after 10 epochs, and the test loss reaches a value of 0.11 at that point. The same graph reveals that after 10 epochs, test accuracy achieves a maximum of 0.95 and train accuracy reaches a maximum of 0.98. CAM generated from AlexNet shows satisfying result as seen in

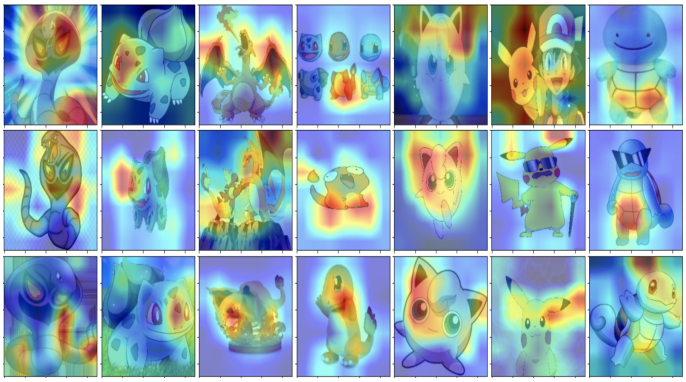

Fig. 6. Class Activation Map using VGG16 CAM network of three different images for each of the seven Pokemons categories. Pokemon categories from left to right - Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle.

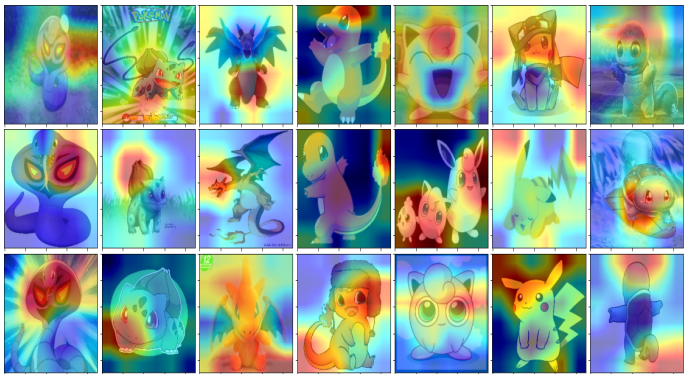

Fig. 8. Class Activation Map using AlexNet CAM network of three different images for each of the seven Pokemons categories. Pokemon categories from left to right - Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle.

Fig. 7, The image grid displays the original Pokemon images from each of the seven categories together with heat maps and blended images that highlight the image's informative area. The image clearly demonstrates how the CAM identifies the head and flamed tail tip of Charmander as crucial regions for prediction, as well as Jigglypuff's face. Fig. 8 shows the informative sections in three distinct images, one for each type of Pokémon. The most important feature in each photograph, the Arbok's eyes, are once again highlighted in red, exactly like the Jigglypuff's forehead and other Pokémon's as well.

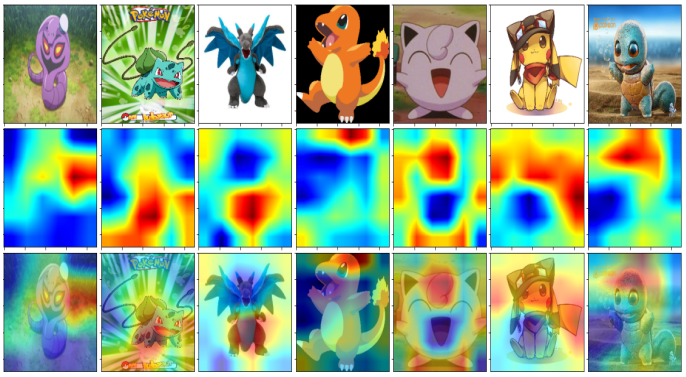

Fig. 7. Class Activation Map of seven categories of Pokemons using Alexnet CAM network. First row- Original Image, second row- Heatmap, third row- Blended image. Pokemon categories from left to right - Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle.

C. VGG19 GRAD-CAM

In order to retrieve the gradient from the final convolution layer, we modified the VGG19 network and trained it on our training dataset. The VGG19 GRAD-CAM network's train and test losses are shown in the bottom right graph of the Fig. 4. The model's train loss decreases from 4 to 2.7 and the test loss reaches a value of 0.98 after 10 epochs. According to the same figure with top right graph, test accuracy peaks at 0.88 and train accuracy peaks at 0.80 after 10 epochs. Classification accuracy of this model comes out to be low as compared to other two networks and can be observed through the output images in Fig.9. The essential region of images are not highlighted by the GRAD-CAM created from the final convolutional layer of the VGG19 network, which results in inaccurate prediction and low accuracy. We can see that the red patches are stretched out toward the corners in the majority of the images.

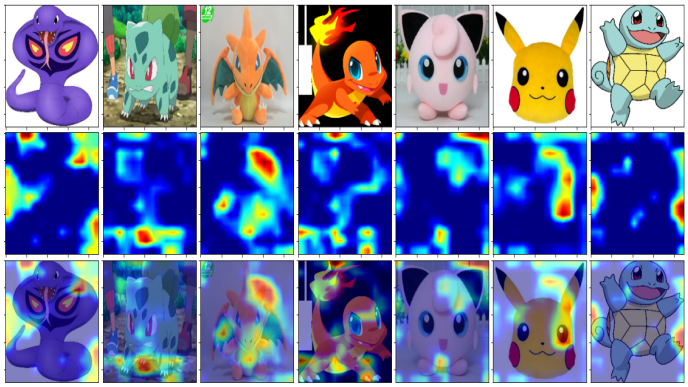

Fig. 9. Class Activation Map of seven categories of Pokemons using VGG19 Grad-CAM network. First row- Original Image, second row- Heatmap, third row- Blended image. Pokemon categories from left to right - Arbok, Bulbasaur, Charizard, Charmander, Jigglypuff, Pikachu, and Squirtle.

D. Overall Results

The output from the three models—VGG16 CAM, AlexNet CAM, and VGG19 GRAD-CAM is shown in Fig. II-B . The plot's first horizontal grid displays the original image of Pokémon drawn from seven different categories. Next, a horizontal grid of blended images created by the VGG16 CAM network is displayed. The third horizontal grid displays the output from the AlexNet CAM, and the last horizontal grid displays the VGG19 GRAD-CAM output. We can see that the major region of each Pokemon shown by both the CAM networks We can see that the areas marked in red for each Pokemon in the VGG16 CAM output and AlexNet CAM output are relevant, such as the Bulbasaur's face and tip of

Charmander's tail. The Fig. 4 shows that the results of the AlexNet CAM are slightly more pleasing than those of the VGG16 CAM. The GRAD-CAM class activation map results are not sufficient to emphasize the pertinent areas of an image that can help with the accurate categorization. Additionally, VGG19 GRAD-precision CAM's was found to be subpar. The goal of our investigation, however, was to comprehend the inner workings of CNN and determine what data it is gathering for classification.

V. Discussion and Summary

In conclusion, the paper discusses about the analysis of Class Activation Maps using Deep Convolutional Neural Networks for Pokémon classification. This work allows us to find out and visualize the major region of an image used in prediction. By substituting the fully-connected layer with the global average pooling layer, we were able to create the Class Acivation Maps using the VGG16 network and AlexNet network. Our research demonstrates that while both VGG16 CAM and AlexNet CAM accurately classify the photos with high accuracy and display the significant region of images for each category, AlexNet slightly beats VGG16. To create the class activation maps, we applied the GRAD-CAM approach to the VGG19 network and extracted the gradient value from the final convolutional layer. As you can see in Fig.2 the GRAD-CAM results do not appear to be satisfactory; rather than highlighting the discriminative region of an image, the red region that is supposed to represent the key portion of an image instead focuses on the corners of the images. Also, the accuracy from this model was comparatively low. As previously indicated, CAM can be utilized to improve the model performance by examining the heatmaps and identifying which aspect of the image is confounding the network and causing incorrect predictions, which in turn leads to decreased accuracy. To improve classification accuracy and the ability to distinguish between different sections of an image, we can attempt to improve the GRAD-CAM network in future work by modifying the necessary parameters or, if necessary, layers.

VI. Acknowledgment

We would like to show our sincere gratitude towards Dr.Bruce Maxwell, our professor at Northeastern University, for the past twelve weeks of teaching us the concepts of computer vision which we have intensively applied throughout this project. The way he taught us Convolutional Neural Networks and introduced us to the class activation map which are the cornerstones of this project, enabled us to complete the work. We would also like to thank all the experts who have contributed greatly in this space and whose papers [1] have given us inspiration for this project.

References

[1] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva and A. Torralba, "Learning Deep Features for Discriminative Localization," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2921-2929, doi: 10.1109/CVPR.2016.319.

[2] J. R. Lee, S. Kim, I. Park, T. Eo and D. Hwang, "Relevance-CAM: Your Model Already Knows Where to Look," 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, pp. 14939-14948, doi: 10.1109/CVPR46437.2021.01470.

[3] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618-626, doi: 10.1109/ICCV.2017.74.

[4] Jung, Hyungsik Oh, Youngrock. (2021). LIFT-CAM: Towards Better Explanations for Class Activation Mapping.

[5] P. -T. Jiang, C. -B. Zhang, Q. Hou, M. -M. Cheng and Y. Wei, "LayerCAM: Exploring Hierarchical Class Activation Maps for Localization," in IEEE Transactions on Image Processing, vol. 30, pp. 5875-5888, 2021, doi: 10.1109/TIP.2021.3089943.

[6] P. Piedra, C. Gobert, A. Kalume, P.Y. Le, M. Kocifaj, K. Muinonen, et al, "Where is the machine looking? Locating discriminative light-scattering features by class-activation mapping", J Quant Spectrosc Radiat Transf, 247 (2020), Article 106936, 10.1016/j.jqsrt.2020.106936

[7] T. Guo, J. Dong, H. Li and Y. Gao, "Simple convolutional neural network on image classification," 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), 2017, pp. 721-724, doi: 10.1109/ICBDA.2017.8078730.